

# Learning Force Field Parameters from Differentiable Particle-Field Molecular Dynamics

Manuel Carrer, Henrique Musseli Cezar, Sigbjørn Løland Bore, Morten Ledum,  
and Michele Cascella\*

*Hylleraas Centre for Quantum Molecular Sciences  
and Department of Chemistry,  
University of Oslo,  
PO Box 1033 Blindern, 0315 Oslo, Norway*

E-mail: [michele.cascella@kjemi.uio.no](mailto:michele.cascella@kjemi.uio.no)

## Abstract

We develop  $\partial$ -HylleraasMD ( $\partial$ -HyMD), a fully end-to-end differentiable molecular dynamics software based on the Hamiltonian hybrid particle-field formalism, and use it to establish a protocol for automated optimization of force field parameters.  $\partial$ -HyMD is templated on the recently established HylleraasMD software, while using the JAX autodiff framework as the main engine for the differentiable dynamics.  $\partial$ -HyMD exploits an embarrassingly parallel optimization algorithm by spawning independent simulations, whose trajectories are simultaneously processed by reverse mode automatic differentiation to calculate the gradient of the loss function, which is in turn used for iterative optimization of the force-field parameters. We show that the parallel organization facilitates the convergence of the minimization procedure, also avoiding the known memory and numerical stability issues of differentiable molecular dynamics approaches. We showcase the effectiveness of our implementation by producing a library of force field parameters for standard phospholipids, with either zwitterionic or anionic heads, and with saturated or unsaturated tails. Compared to the all-atom reference, the force field obtained by  $\partial$ -HyMD yields better density profiles than the parameters derived from previously utilized gradient-free optimization procedures. Moreover,  $\partial$ -HyMD models can with good accuracy predict properties not included in the learning objective, such as lateral pressure profiles, and are transferable to other systems such as triglycerides.

# Introduction

Coarse-grained modeling enables the simulation of systems on length and time scales that are orders of magnitude larger than those accessible in traditional all-atom simulations.<sup>1</sup> By adopting a low-resolution representation of molecules and computationally efficient interaction potentials, coarse-graining is associated with a smoothening of the molecular free-energy landscape.<sup>2</sup> Due to the contraction of the phase-space, dynamics of coarse-grained models is intrinsically accelerated, providing additional speedup and faster statistical sampling over all-atom simulations, beyond the increased speed due to lower-resolution molecular representation and fast effective interaction potentials,<sup>3</sup> even though this comes at the cost of a generally poor description of dynamical properties, such as diffusion coefficients, unless explicitly accounted for.<sup>4</sup> Acceleration in the conformational sampling enables, for example, the straightforward study of phase separation, like the self-assembly of lipid bilayers,<sup>5,6</sup> or phase transitions, like spontaneous ice nucleation.<sup>7</sup>

Particle-field models represent indirectly the interactions between different molecules through by their density fields, leading to a particularly high level of smoothening of the free-energy surface and hence a correspondingly large acceleration of the dynamics.<sup>8</sup> In fact, we reported that particle-field molecular dynamics models enable (sub-)nanosecond equilibration of dispersed charged surfactants to fully self-assembled structures even in the presence of large activation energies<sup>9,10</sup> that would significantly lengthen the characteristic self-assembling times even using particularly successful CG models like the MARTINI.<sup>11–13</sup> As such, particle-field approaches provide a powerful tool for studying processes that are computationally inaccessible by standard molecular modeling.

While particle-field simulations are maturing both in terms of mathematical foundations<sup>14,15</sup> and open-source software implementation,<sup>9,16,17</sup> the parameterization of the particle-field force field remains challenging. On top of being coarse-grained, which targets the more elusive potential of mean force instead of the potential energy surface of all-atom simulations, the determination of the non-standard particle-field interactions is

not straightforwardly amenable to traditional techniques, such as iterative Boltzmann inversion or force-matching.<sup>18,19</sup> The key challenge resides in calibrating the  $\tilde{\chi}_{kl}$ -parameter matrix, which describes the mixing energy between the density of two species  $k$  and  $l$ . Earlier works investigating polymeric systems typically considered a confined range of  $\tilde{\chi}$ -parameters to understand the nature of their phase diagram.<sup>8</sup> In subsequent works targeting biological lipids, the  $\tilde{\chi}$ -parameters were chosen by a combination of inference from MARTINI Lennard-Jones parameters and by-hand adjustments in order to reproduce all-atom density profiles.<sup>20</sup> To systematize the parameterization of  $\tilde{\chi}$ -parameters, some of us proposed an automated machine-learning procedure.<sup>21</sup> There, similar to the self-driving labs approach in the Guzik Group,<sup>22</sup> Bayesian Optimization (BO) was used to run simulation experiments that compute density profiles for different  $\tilde{\chi}$ -values, determining the optimal  $\tilde{\chi}$ -parameters that maximize the agreement with all-atom reference profiles. The approach is very general and has the advantage over bottom-up coarse-graining approaches in that it can incorporate both experimental and all-atom data, as exemplified in a recent work, where the loss function to parameterize dipalmitoyl-phosphatidylcholine (DPPC) phospholipid bilayers included both the computationally-determined density profiles and the experimental value of the area-per-lipid.<sup>23</sup>

BO provides a state-of-the-art algorithm for optimizing costly fitness functions without access to gradients in low-dimensional parameter spaces.<sup>24</sup> However, this approach becomes problematic when the number of particle species  $S$  increases, resulting in a large  $S(S - 1)/2$ -dimensional parameter space, or, even worse when also including intramolecular bonded parameters into the optimization problem. Optimization techniques employing gradients generally achieve better convergence and work well on higher-dimensional spaces; therefore, they are better-suited for force field parameterization. In this regard, differentiable molecular dynamics, leveraging advancements in GPU-based hardware, highly efficient implementations of automatic differentiation (autodiff),<sup>25</sup> and a generalized back-propagation algorithm,<sup>26</sup> is becoming increasingly popular. Differentiable

molecular dynamics can be used to optimize force field parameters through the dynamical evolution of the systems under study. A conventional machine-learning based approach consists in training artificial neural network (NN) interatomic potentials on high-accuracy data (for example, either DFT calculation datasets, or all-atom simulations for CG modeling).<sup>27,28</sup> This methodology has been successfully employed to determine NN pair potentials for CG water models that replicates radial<sup>29,30</sup> and angular distribution<sup>31</sup> functions, a NN potential for a CG model of chignolin<sup>32</sup> and the parameterization of a CG protein force field, where the parameters are trained to minimize the root-mean-square deviation (RMSD) from the native state structures.<sup>33</sup> Differentiable molecular dynamics however is not only limited to training NN potentials, but in principle can also be applied to the optimization of known analytical functions, like those that are still commonly used in molecular simulations.

Given these successes, in the following, we present  $\partial$ -HylleraasMD ( $\partial$ -HyMD), a re-implementation of the HylleraasMD (HyMD)<sup>9,17</sup> code for hamiltonian hybrid particle-field (HhPF) molecular dynamics in the JAX<sup>34,35</sup> differentiable framework, that allows us to perform general force field parameterization on *any* target observable that depends on the simulation trajectory. Our main goal is to optimize the intermolecular interactions described by HhPF. As a test case, we apply our optimization protocol on a diverse set of lipids, providing a library of parameters that can be used for future studies. We also test the performance of these parameters for the self-assembly of lipid bilayers and the transferability to similar systems such as triglycerides.

## Methods

### Hamiltonian hybrid particle-field dynamics

We start by briefly summarizing the Hamiltonian hybrid particle-field molecular dynamics approach,<sup>9,14</sup> for which parameters will be optimized. Consider a system of  $M$  molecules,

with the  $i^{th}$  molecule containing  $N_i$  particles at positions  $\mathbf{R}_i = \{\mathbf{r}_j\}_{j=1}^{N_i}$  with conjugate momenta  $\mathbf{P}_i = \{\mathbf{p}_j\}_{j=1}^{N_i}$ , subject to the Hamiltonian<sup>14</sup>

$$\mathcal{H}(\mathbf{R}, \mathbf{P}) = \sum_{i=1}^M \mathcal{H}_0(\mathbf{R}_i, \mathbf{P}_i) + W[\{\tilde{\phi}(\mathbf{r})\}], \quad (1)$$

with

$$\mathcal{H}_0(\mathbf{R}_i, \mathbf{P}_i) = \sum_{j=1}^{N_i} \frac{\mathbf{p}_j^2}{2m_j} + \mathcal{U}_0(\mathbf{R}_i). \quad (2)$$

Here  $\mathbf{R}$  and  $\mathbf{P}$  denote the collection of positions and momenta of all particles,  $\mathcal{H}_0$  is the non-interacting Hamiltonian for the single  $i^{th}$  molecule,  $\mathcal{U}_0$  is the intramolecular potential energy (containing only bonded terms) and  $W[\{\tilde{\phi}\}]$  is the interaction energy functional, dependent on the densities of the particles  $\{\tilde{\phi}\}$ , modeling the intermolecular interactions. We employ the shorthand  $\{\tilde{\phi}\} = \{\tilde{\phi}_\ell\}_{\ell=1}^S$  for the collection of number densities associated with all  $S$  different particle species in the system. This hybrid approach aims to achieve molecular resolution through  $\mathcal{H}_0$  and a smooth free energy landscape by adopting a density-dependent interaction energy functional, with minimal steric hindrance. Since  $\mathcal{H}_0$  is nothing but a standard molecular mechanics Hamiltonian, in the following we focus on  $W[\{\tilde{\phi}\}]$ . We adopt a variation of the commonly used Flory-Huggins interaction functional:

$$W[\{\tilde{\phi}(\mathbf{r})\}] = \frac{1}{2\rho_0} \int d\mathbf{r} \sum_{\ell m} \tilde{\chi}_{\ell m} \tilde{\phi}_\ell(\mathbf{r}) \tilde{\phi}_m(\mathbf{r}) + \frac{1}{2\kappa\rho_0} \int d\mathbf{r} \left( \sum_{\ell} \tilde{\phi}_\ell(\mathbf{r}) - a \right)^2, \quad (3)$$

where  $\rho_0$  is defined as the density of a coarse-grained particle,  $\kappa$  is the compressibility, which controls the local fluctuations of the density,  $a$  is a free parameter that can be tuned to calibrate the correct average density at the target temperature and pressure of interest and finally  $\tilde{\chi}_{\ell m}$  are the mixing interaction parameters between particle densities of types  $\ell$  and  $m$ . These parameters are the optimization targets for our present study.

The sampling of Eq. 3 can be achieved in multiple ways, such as in Monte Carlo, often

referred to as the Single-Chain-In-Mean-Field method,<sup>8</sup> or by molecular dynamics with various formulations.<sup>9,17,36,37</sup> We adopt our recently developed Hamiltonian approach, which is the only implementation demonstrated to achieve energy-conserving and alias-free dynamics.<sup>14</sup> In brief, this approach builds on standard particle-mesh operations<sup>38</sup> by assigning particle number densities onto a regular grid via an assignment function  $P$  and subsequently performing a convolution with a filter function  $\mathcal{G}$  that defines the density spread associated with particle species  $t$  as

$$\tilde{\phi}_t(\mathbf{r}) \equiv \int \phi_t(\mathbf{x}) \mathcal{G}(\mathbf{r} - \mathbf{x}) d\mathbf{x}, \quad \text{with} \quad \phi_t(\mathbf{r}) = \sum_{i=1}^{N_t} P(\mathbf{r} - \mathbf{r}_i). \quad (4)$$

The force on a particle placed at  $\mathbf{r}_i$  is then obtained by the direct spatial derivative of this interaction energy functional as

$$\mathbf{F}_i = -\frac{\partial W}{\partial \mathbf{r}_i}. \quad (5)$$

The above can be recast, as worked out in ref.<sup>14</sup>, into the form:

$$\mathbf{F}_i = -\int \nabla V(\mathbf{r}) P(\mathbf{r} - \mathbf{r}_i), \quad \text{with} \quad V(\mathbf{r}) = \frac{\delta W}{\delta \phi(\mathbf{r})}, \quad (6)$$

where  $V(\mathbf{r})$  is called the *external potential*, and its gradients are obtained numerically by Fast Fourier Transform (FFT) operations.<sup>14</sup>

## Differentiable molecular dynamics

Our main objective is to minimize an arbitrary loss function  $\mathfrak{L}$ , which depends on an observable  $A$ , with respect to a set of  $T$  force field parameters  $\theta = \{\Theta^i\}_{i=1}^T$ ,

$$\theta_{\text{optimal}} = \arg \min_{\theta} \mathfrak{L}(\langle A(U_{\theta}) \rangle), \quad \text{where } \theta \in \mathbb{R}^T. \quad (7)$$

Here the angle brackets denote the ensemble average, while  $U_\theta = U_\theta(\mathbf{R}; \Theta_1, \Theta_2, \dots, \Theta_T)$  is the potential energy depending parametrically on  $\theta$ . Assuming ergodicity, the ensemble average of the observable  $A$  is calculated as the time average over a trajectory composed of  $N_f$  frames by

$$\langle A(U_\theta) \rangle = \frac{1}{N_f} \sum_{i=1}^{N_f} A(\mathbf{R}^i), \quad (8)$$

where superscript  $i$  denotes positions and velocities of all particles,  $\mathbf{R}^i$  and  $\mathbf{V}^i$  respectively, at discrete time step  $i$ . The particle positions are evolved in time through a discrete and recursive update step:

$$\mathbf{R}^i = \mathbf{R}^{i-1} + f(\mathbf{R}^{i-1}, \mathbf{V}^{i-1}; \theta) \Delta t, \quad (9)$$

where  $\Delta t$  the time step and  $f$  denotes the update scheme. In our implementation we use the rRESPA integrator,<sup>39</sup> where the positions-update step is the same as in the velocity-Verlet algorithm. Therefore, we have

$$f(\mathbf{R}^i, \mathbf{V}^i; \theta) = \mathbf{V}^i + \frac{\Delta t}{2m} \mathbf{F}(\mathbf{R}^i; \theta), \quad (10)$$

where  $\mathbf{F} = -\nabla U_\theta$  are the forces acting on the particles, which directly depend on the force-field parameters. By tracing the update step, we can find the corresponding gradient of  $\mathcal{L}$  with respect to the force field parameters by recursively applying the chain rule:

$$\nabla_\theta \mathcal{L} = \frac{d\mathcal{L}}{d\theta} = \sum_{i=1}^N \frac{d\mathcal{L}}{d\mathbf{R}^i} \frac{d\mathbf{R}^i}{d\theta} \quad (11)$$

$$= \sum_{i=1}^N \frac{d\mathcal{L}}{d\mathbf{R}^i} \frac{d}{d\theta} (\mathbf{R}^{i-1} + f(\mathbf{R}^{i-1}, \mathbf{V}^{i-1}; \theta) \Delta t) \quad (12)$$

$$= \sum_{i=1}^N \frac{d\mathcal{L}}{d\mathbf{R}^i} \sum_{j=0}^{i-1} \frac{d}{d\theta} (f(\mathbf{R}^j, \mathbf{V}^j; \theta)). \quad (13)$$



Finally, with the gradient of the loss function, we can then use any gradient descent optimizer to steer the  $\theta$ -parameters towards the optimum by updating the force field parameters

$$\theta^{i+1} = \theta^i - \eta \mathcal{S}_p(\nabla_{\theta} \mathfrak{L}), \quad (14)$$

where  $\eta$  is the learning rate and  $\mathcal{S}_p$  represents the update step function of the given optimizer. Various packages implementing differentiable molecular dynamics are available, such as JAX-MD,<sup>40</sup> TorchMD,<sup>32</sup> and Molly.<sup>41</sup> However, these programs lack some critical features, especially for what concerns particle mesh routines that are needed for the HhPF non-bonded interactions. Therefore, we have implemented our differentiable framework in  $\partial$ -HyMD based on our open-source HhPF simulator HyMD.<sup>9,17</sup> In  $\partial$ -HyMD we use JAX<sup>34,35</sup> to trace the update steps and get the gradient of the loss function as in Equation 13, while implementing the HhPF operations with fast Fourier transforms, the MD integrator, barostat and thermostat using JAX NumPy API, and taking advantage of JIT compilations whenever possible.

## Differentiable density profiles

In order to compute its derivatives, the loss function needs to be continuously differentiable. For the purpose of this work, the property that we want to target is the membrane lateral density profile, which is typically obtained by computing the histogram of the  $z$  coordinate of the particles (assuming here that  $z$  is the direction normal to the membrane). To calculate a differentiable density profile we can approximate the histogram with a simple Gaussian kernel density estimation, where we center a normal distribution around each discrete value  $z_i$ :

$$\hat{\rho}_t(z) \simeq \frac{\Delta z}{N_T} \sum_{i=1}^{N_T} \frac{1}{\sqrt{2\pi}h^2} e^{-\frac{(z-z_i)^2}{2h^2}}. \quad (15)$$

Here  $N_T$  is the total number of particle of type  $T$ ,  $\Delta z$  is the bin width and  $h$  is the Gaussian bandwidth. To compare with the all-atom number densities, we need to take care of the

normalization, by remembering that the Gaussians are already normalized to one. Therefore, the final density profile is given by:

$$\rho_t(z) = \frac{\hat{\rho}_k(z)N_T}{L_x L_z n_b \Delta z} \simeq \frac{1}{L_x L_y n_b} \sum_{i=1}^{N_T} \frac{1}{\sqrt{2\pi h^2}} e^{-\frac{(z-z_i)^2}{2h^2}}, \quad (16)$$

where  $L_x$  and  $L_y$  are the box lengths along the respective axis, and  $n_b$  is the number of bins.

## Optimization protocol

Differentiable MD has two main drawbacks. Firstly, the memory complexity of the optimization scales linearly both with the number of particles and the number of simulation steps, due to the requirement of tracing all the operations in the MD algorithm. Secondly, by increasing the number of steps, one might incur into so-called gradient explosion, where gradient sums in Eq. 13 grow instantly larger, leading to disruptive numerical instability. However, to obtain a well-sampled property, an adequate number of frames is needed. To the best of our knowledge, only one work<sup>31</sup> in the literature suggests an approach, using trajectory reweighing via Boltzmann distributions, that tries to alleviate the second problem. In this work, we opt for a parallel setup, as schematized in Fig. 1.

The main idea is to spawn independent replica simulations starting from the same initial CG structure and the same set of starting  $\chi$  parameters, but with different random number generator seeds. We call this protocol parallel replica simulations (PRS). Initially, we run some equilibration, where we let the system evolve without tracing any operation. This is followed by a shorter differentiable MD simulation to obtain the target property. Our loss

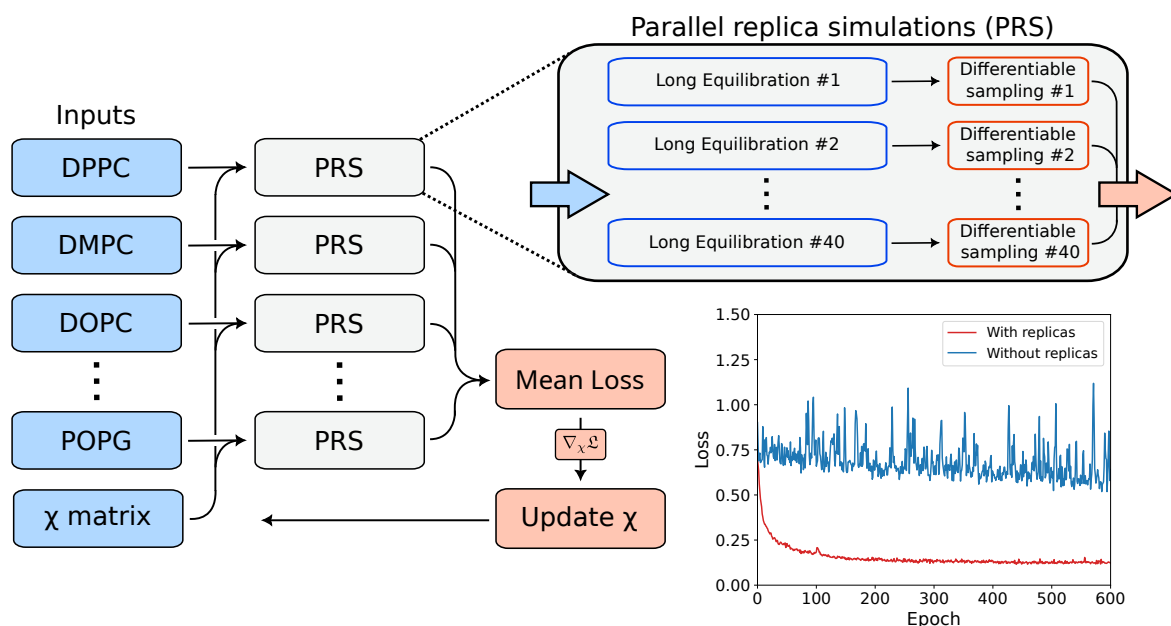


Figure 1: Differentiable MD protocol for the optimization of the lipid library. On the left, overall training architecture, in which each lipid runs a parallel replica simulation (PRS) without any communication. The results are combined to compute the mean loss (Equation 19) which is the quantity effectively minimized. In the zoomed-in part at the top right, a single lipid optimization unit – PRS – which is composed of independent parallel simulations containing an equilibration followed by differentiable sampling. Each PRS unit produces a loss function defined by Equation 18. On the bottom right, comparison between the losses obtained from training a single DPPC membrane, with and without replicas.

function is then computed as the mean of the loss functions over the replicas:

$$\mathcal{L}_{\chi_i}^{\text{PRS}} = \frac{1}{N_R} \sum_{n=1}^{N_R} \mathcal{L}_n(\chi_i) \quad (17)$$

$$= \frac{1}{N_R} \sum_{n=1}^{N_R} \left( \frac{1}{n_T} \sum_{t=1}^{n_T} \frac{w_t}{n_b} \sum_{b=1}^{n_b} (\rho_t(z_b) - \rho_t^{\text{ref}}(z_b))^2 + w_a (A_L - A_L^{\text{ref}})^2 + \frac{1}{2} \sum_{l \neq m} \left| \frac{\tilde{\chi}_{l,m}}{\Delta \chi} \right|^3 \right), \quad (18)$$

where  $N_R$  is the number of replicas. The inner loss function is composed of three terms. The first one targets the density profiles. In this case,  $n_T$  is the number of particle types in the system,  $n_b$  is the number of bins,  $w_t$  is an optional weight parameter, and the densities  $\rho_t$  and  $\rho_t^{\text{ref}}$  are estimated at the bin centers  $z_b$ . The second one targets the area per lipid (APL in the text, hereafter).  $A_L$  is the mean APL, naively calculated from the simulations

as  $\frac{2L_x L_y}{N_{\text{lipids}}}$ ,  $A_L^{\text{ref}}$  is the experimental value for the APL, and  $w_A$  is another optional weight. The last term is a cubic constraint introduced to prevent training parameters that would lead to unstable simulations, by limiting the numerical range of the  $\tilde{\chi}_{l,m}$  matrix elements. Here,  $\Delta\chi$  is the boundary value.

In Fig 1 we also show the difference between the loss function obtained by training a DPPC membrane with and without the PRS method. Without PRS, the loss is noisy and slowly decreasing, meaning that, due to the short single differentiable MD simulation, we are not able to consistently learn the parameters.

Since training single lipid membranes on their own would lead to non-transferable overfitted parameters to that specific lipid (see Supporting Information), in each epoch we train a dataset of them, and apply the parameter update step only after all the lipids in the set have been simulated (Fig. 1A). This ensures that the parameters are transferable among the different lipids. We achieve this by computing a mean loss

$$\mathfrak{L}_{\chi_i}^{\text{mean}} = \frac{1}{N_{\text{lipids}}} \sum_{j=1}^{N_{\text{lipids}}} \mathfrak{L}_{\chi_i}^j \quad (19)$$

where  $\mathfrak{L}_{\chi_i}^j$  is the PRS loss for the  $j$ -th of the  $N_{\text{lipids}}$  optimized simultaneously, defined by Equation 18. The mean loss is what is effectively minimized and gives the update step for the force field parameters (Equation 14).

## Computational details

All-atom membrane systems for different lipids were set up using CHARMM-GUI,<sup>42</sup> namely, for 1,2-dipalmitoyl-sn-glycero-3-phosphocholine (DPPC), 1,2-dimyristoyl-sn-glycero-3-phosphocholine (DMPC), 1,2-dioleoyl-sn-glycero-3-phosphocholine (DOPC), 1-palmitoyl-2-oleoyl-glycero-3-phosphocholine (POPC), 1,2-dimyristoyl-sn-glycero-3-phosphoethanolamine (DMPE), 1,2-dioleoyl-sn-glycero-3-phosphoethanolamine (DOPE), 1-palmitoyl-2-oleoyl-glycero-3-phosphoethanolamine (POPE), 1,2-dipalmitoyl-sn-glycero-3-

phosphoglycerol (DPPG), 1,2-dimyristoyl-sn-glycero-3-phosphoglycerol (DMPG), 1,2-dioleoyl-sn-glycero-3-phosphoglycerol (DOPG), and 1-palmitoyl-2-oleoyl-glycero-3-phosphoglycerol (POPG). The box size was approximately  $5 \times 5 \times 8$  nm and the total number of lipid molecules was  $\sim 80$ , depending on the system. After equilibration, all atomistic MD simulations were run for 100 ns, using a time step of 2 fs, with the CHARMM-36m<sup>43</sup> force field in GROMACS 2021.5.<sup>44,45</sup> Temperature control above the melting temperature  $T_m$  of each lipid was ensured by using the CSVR<sup>46</sup> thermostat, with  $\tau_T = 1$  ps, and the pressure was maintained at 1 bar using the semi-isotropic cell rescale barostat,<sup>47</sup> with  $\tau_P = 5$  ps. PME was used to compute long-range electrostatics, with a real space cut-off radius of 1.2 nm. The same cut-off was also used for the Lennard-Jones interactions, and bonds involving hydrogen atoms were constrained with LINCS.<sup>48</sup>

The all-atom trajectories were then coarse-grained using the PyCGTOOL package<sup>49</sup> and the MARTINI 2 mapping.<sup>50</sup> The mapping and bead types used for the parametrization are shown in Fig. 2, where beads represented by the same color in different lipids have the same  $\chi$  parameters. The bond distances were calculated from the average distances in the atomistic trajectories (see Table S2 in the Supporting Information), while the reference angles for the three-body interatomic potential and all harmonic force constants were taken from MARTINI. We computed the reference lateral density profiles with MDAnalysis<sup>51,52</sup> from these CG-mapped trajectories, and used them in the differentiable MD optimization. For what concerns the parallel replica CG simulations, the equilibration was run for 2000 steps (200 ps) while the differentiable MD for 200 steps (20 ps). Since we use 40 replicas for each optimization, the gradients are effectively computed from  $40 \times 20$  ps = 800 ps long simulations. In both equilibration and production, we used a time step of 0.02 ps for the interatomic forces calculation and a time step of 0.1 ps for the intermolecular and electrostatic forces. The systems were kept at constant temperature with the velocity rescale thermostat<sup>46</sup> and at constant pressure with the semi-isotropic Berendsen barostat<sup>53</sup>. The coupling constant was set to 0.1 ps for both. We used a  $20 \times 20 \times 30$  grid

for the particle mesh calculations, compressibility  $\kappa = 0.05 \text{ kJ}^{-1} \text{ mol}$ , particle spread  $\sigma = 0.5 \text{ nm}$ ,  $\rho_0 = 8.33 \text{ nm}^{-3}$  and  $a = 9.21 \text{ nm}^{-3}$ .

Training was carried out over 600 epochs. We set  $w_t = 1 \text{ nm}^3$ ,  $w_A = 3 \text{ nm}^{-2}$ , and  $\Delta\chi = 300 \text{ kJ mol}^{-1}$ , while for the  $\chi$  parameter updates, we used the AdaBelief optimizer<sup>54</sup> with  $b_1 = 0.1$ ,  $b_2 = 0.4$ , and learning rate  $\eta = 0.01$ . To validate the trained parameters, we prepared bigger simulation boxes ( $10 \times 10 \times 10 \text{ nm}^3$ ,  $\sim 380$  lipids) for all the systems. For these systems, we used a  $40 \times 40 \times 40$  grid, while keeping all the other simulation parameters unchanged, and ran the simulations for 50 ns. For the DPPC self-assembly validation, we started from the big simulation box system and scrambled the lipids by running a simulation at high temperature, 500 K. Then to obtain the self-assembled membrane, we ran an NVT simulation at 323 K. Finally, we investigated the phase separation of triolein, a triglyceride (TG), in a DOPC bilayer. The CG mapping of triolein was taken from Ref<sup>55</sup>. We used one of the setups provided in Ref.<sup>56</sup>, where the membrane is composed of 3200 DOPC molecules and 320 TGs. In this case, the box is approximately  $33 \times 33 \times 17 \text{ nm}^3$  big, and contains  $\sim 113\,000$  water beads.

## Results

We use  $\partial$ -HyMD and the outlined optimization protocol to obtain the  $\tilde{\chi}_{\ell m}$  matrix for the non-bonded interaction between each bead in the phospholipid set. The simulations were carried out at temperatures in which the lipids are in the fluid lamellar phase, and for which we could find experimental data for the APL in the literature. The temperatures and experimental APLs are shown in Table 1. Other than the optimization procedure, compared to the previous models, we use different bond distances. This decision was taken during the initial testing of the optimization protocol, by noticing that the head group peaks would systematically appear less hydrated compared to the respective CG-mapped all-atom peaks. By further tweaking the angle parameters, it might be possible to fully recover the all-atom

description of the membrane core. However, for the purpose of this work, we keep the  $180^\circ$  angles in our parametrization, as changes in the alkyl angles may reflect in the melting points.<sup>57</sup> The bonded parameters are reported in Table S2 of the Supporting Information.

We managed to optimize the parameters for the set in 600 epochs, as shown in Fig. 2. The

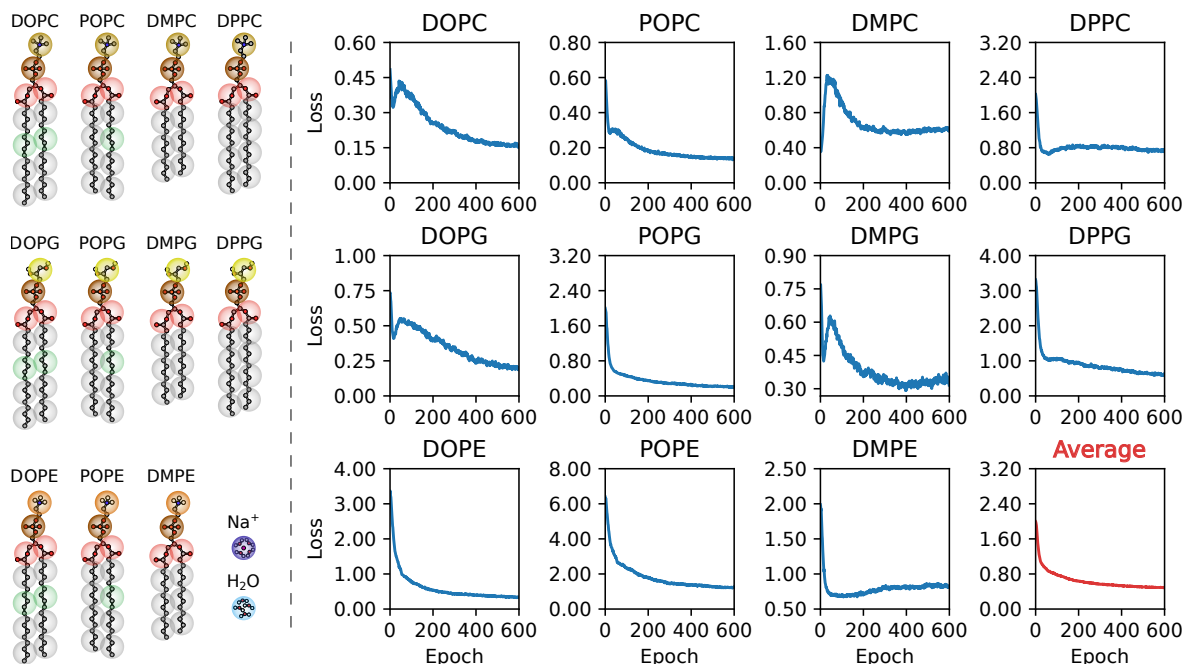


Figure 2: On the left, all-atom to coarse-grained bead mapping. The different bead colors represent the particle types for which the intermolecular interactions were trained in this work. The phosphate groups (represented in brown) are negatively charged ( $-1$ ), the PC and PE head groups and the  $\text{Na}^+$  ions have positive charge ( $+1$ ). Apart from the PG and PE head groups, hydrogen atoms were omitted in the all-atom representation. On the right, training loss evolution for the set of phospholipids. Individual losses of each PRS (Equation 18) are shown. The panel “Average” contains the mean loss across all the systems, which is the quantity minimized during the optimization procedure (Equation 19).

mean loss quickly decays from 2 to 0.5. For most systems, the individual loss value decays in the first 400 epochs and then oscillates around a minimum, usually well below 1. For DMPC, we can see that the loss increases in the first 100 epochs, and only then descends, indicating that the starting parameters were already good for this particular system. However, these parameters were not transferable to other systems, especially in the PG and PE series, which show starting losses above 1.

The APLs evolution in Fig. 3 shows how areas converge towards the target experimental APLs. These curves are in most cases highly correlated to the loss evolution curves, due to the square difference penalty in the loss function shown in Equation 18. For some systems (DOPC, POPC, DOPG, POPG, DMPG, DOPE, DMPE) we obtain very good agreements with the experimental values, while in other cases (DMPC, DPPC, DPPG, POPE) the APL curves seem to diverge. Generally, the mean absolute error is  $0.02 \text{ nm}^2$ , corresponding to an average 3.8 % deviation from the experimental reference. In the worst case (POPE), the absolute error is around  $0.04 \text{ nm}^2$ , ( $\approx 7\%$  deviation). These values are, however, within the variability found for different experiments performed with the same lipid at similar conditions. For example, in a review, Nagle *et al.*<sup>58</sup> report values between  $0.63 \text{ nm}^2$  and  $0.69 \text{ nm}^2$  for DPPC at 323 K, compared to the  $\sim 0.63 \text{ nm}^2$  in Petrache *et al.*<sup>59</sup> and Kucerka *et al.*<sup>60</sup>. The average APL deviation is also just slightly larger than the values obtained in a recent refinement of the MARTINI 3 force field, which are up to 4 % in their worst cases.<sup>61</sup> Even though the study of Empereur-mot *et al.*<sup>61</sup> also targeted the APL as one of the observables, the authors used different mappings and focused on optimizing the intramolecular bonded terms, which were not the focus of our optimization.

Despite the loss curves being modulated by the APL, the density profiles also play an important role during the optimization. The density profiles computed in epoch 0, i.e., using the initial set of parameters, and epoch 600 are shown in Fig. 4. We observe a systematic improvement of the density profiles for all the phospholipids, being able to reproduce in most cases peak positions, water penetration, and even structuring of the alkyl tails as seen, e.g., for DOPC, DOPE, and POPE. For the DM and DP series, even with the improvements brought by the optimization, we observe that the intricacies of the alkyl density profile were not perfectly reproduced. We attribute this to the fixed topology adopted during the optimization, with bond and angle parameters not being optimized. As shown in previous studies, optimizing the bonded parameters can greatly improve the description of coarse-grained potentials.<sup>61</sup> However, in this work, we focus on optimizing transferable nonbonded



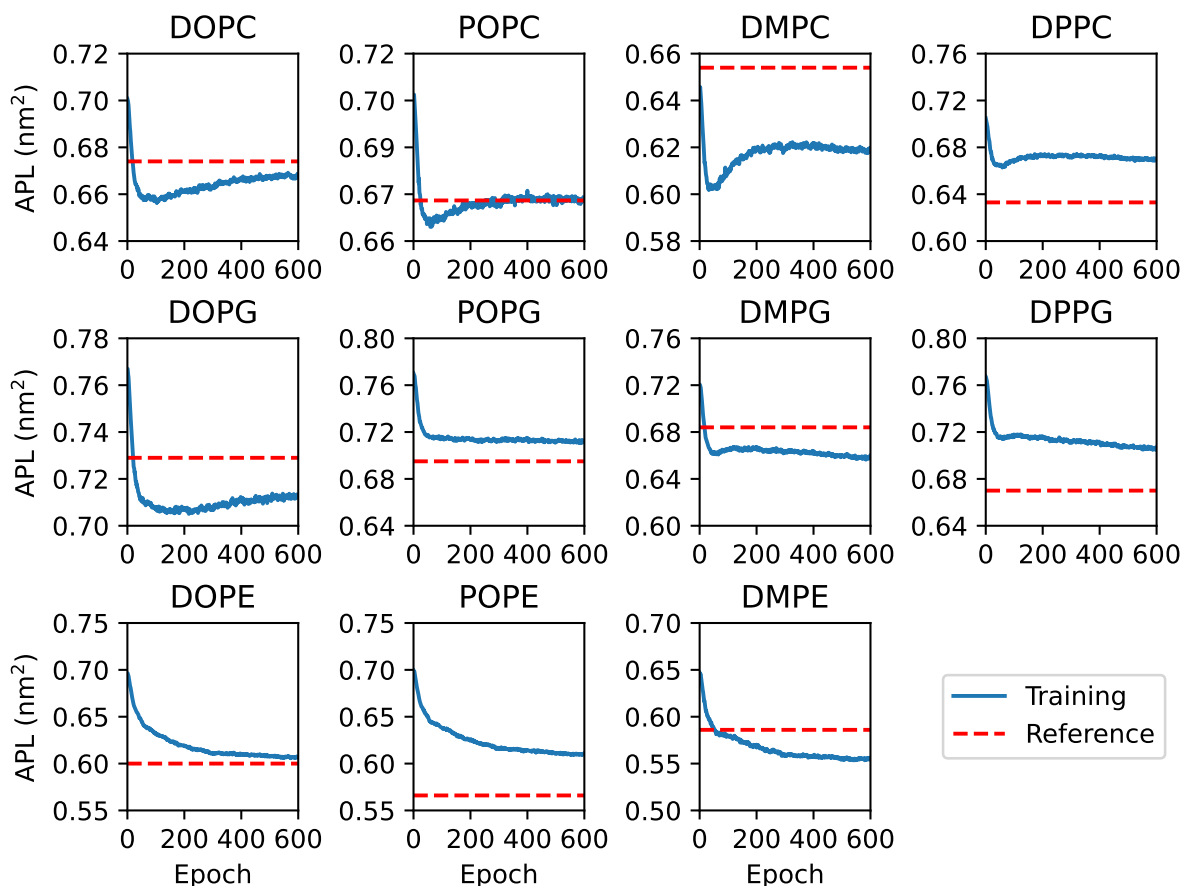


Figure 3: Area per lipid evolution for the set of phospholipids. The horizontal red dashed lines represent the target APL for the system.

parameters (which were kept constant in Ref.<sup>61</sup>), and leave refinements of topology and bonded parameters to future work.

To validate the parametrization of each phospholipid, we ran 50 ns long simulations using the optimized parameters on bigger systems (twice the size in the  $x, y$  directions) and compared them with the reference all-atom density profiles. These results are shown in Fig. S1 of the Supporting Information. The longer simulations in these bigger systems show that only small differences in the height of the peaks are observed compared to the densities obtained during the short parallel simulations used for the optimization. More importantly, the positions of the peaks for each bead are maintained, indicating that the optimized parameters produce stable simulations.

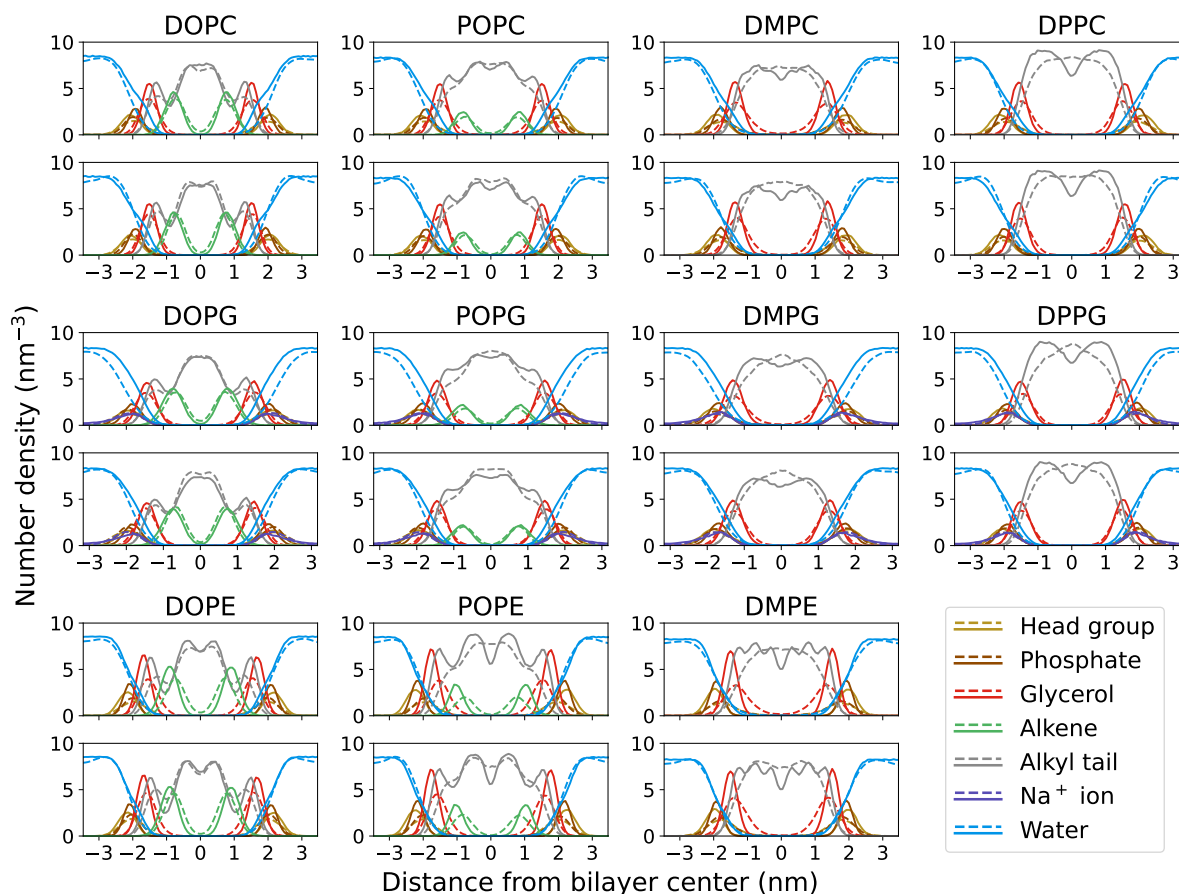


Figure 4: Density profiles for all the optimized lipids (dashed lines), at the start (top) and end (bottom) of training. Solid lines represent the all-atom reference profiles. For simplicity, we use the same color for the head groups, while in the model each lipid family has their own specific bead type.

Outside the differentiable training simulations, good agreement is also maintained between the converged APL and the experimental values, as reported in Table 1. In all cases, the error is around  $0.03 \text{ nm}^2$ . Since the APL was computed naively by considering only the  $xy$  plane area, values slightly below the experimental measures might actually be closer to the reference when membrane undulations are taken into account for the APL calculation. These combined results indicate that the optimized parameters are transferable to larger systems and can be used for production runs.

As a necessary check for the validation of the data sets produced by our optimization setup, we computed the lateral pressure profiles  $\Delta P(z)$  for the optimized lipid bilayers,

Table 1: Reference experimental area per lipid compared to the ones obtained from training, and from 50 ns long HhPD MD simulations of a larger membrane using the learned  $\tilde{\chi}_{\ell m}$  parameters.

System	Experimental (nm <sup>2</sup> )	Training (nm <sup>2</sup> )	Validation (nm <sup>2</sup> )
DPPC	0.633 (50° C) <sup>59</sup>	0.667 ± 0.004	0.664 ± 0.002
DMPC	0.654 (50° C) <sup>59</sup>	0.618 ± 0.004	0.612 ± 0.002
DOPC	0.674 (30° C) <sup>62</sup>	0.669 ± 0.004	0.664 ± 0.002
POPC	0.673 (50° C) <sup>60</sup>	0.674 ± 0.004	0.667 ± 0.002
DMPE	0.586 (60° C) <sup>63</sup>	0.553 ± 0.004	0.550 ± 0.002
DOPE	0.600 (22.5° C) <sup>64</sup>	0.607 ± 0.003	0.603 ± 0.002
POPE	0.566 (30° C) <sup>65</sup>	0.610 ± 0.003	0.608 ± 0.002
DPPG	0.670 (50° C) <sup>66</sup>	0.707 ± 0.004	0.702 ± 0.002
DMPG	0.684 (50° C) <sup>66</sup>	0.660 ± 0.004	0.653 ± 0.002
DOPG	0.729 (50° C) <sup>66</sup>	0.714 ± 0.004	0.708 ± 0.002
POPG	0.695 (50° C) <sup>66</sup>	0.713 ± 0.004	0.708 ± 0.002

defined as:

$$\Delta P(z) = \frac{1}{L_x L_y} \int [P_N(\mathbf{r}) - P_L(\mathbf{r})] dx dy \quad (20)$$

As previously shown using BO-optimized parameters for DPPC in HhPF simulations, the main features of  $\Delta P(z)$  could be reproduced without directly introducing any related information in the learning function.<sup>23</sup> Fig. 5 reports the lateral pressure profile for the same lipid obtained from simulations using parameters by  $\partial$ -HyMD, also in comparison with previously reported data.

Using DPPC for direct comparison, the lateral pressure profile is in excellent quantitative agreement with the one produced by all-atom data, improving the profiles obtained with the BO-optimized parameters. In particular, the lateral pressure profiles feature a weakly negative balance in the membrane core and a large positive fluctuation at the height of the polar head. Notably, in the same region, the stretching, and bending terms provide larger contributions to the pressure unbalance than with the BO model.<sup>23</sup> This is consistent with the sharper distribution of the polar heads reported in the density profiles (Fig. 4), corresponding to a more regular alignment of the bonded moieties along the normal axis of the membrane. Compared to the all-atom model, the pressure profile underestimates the

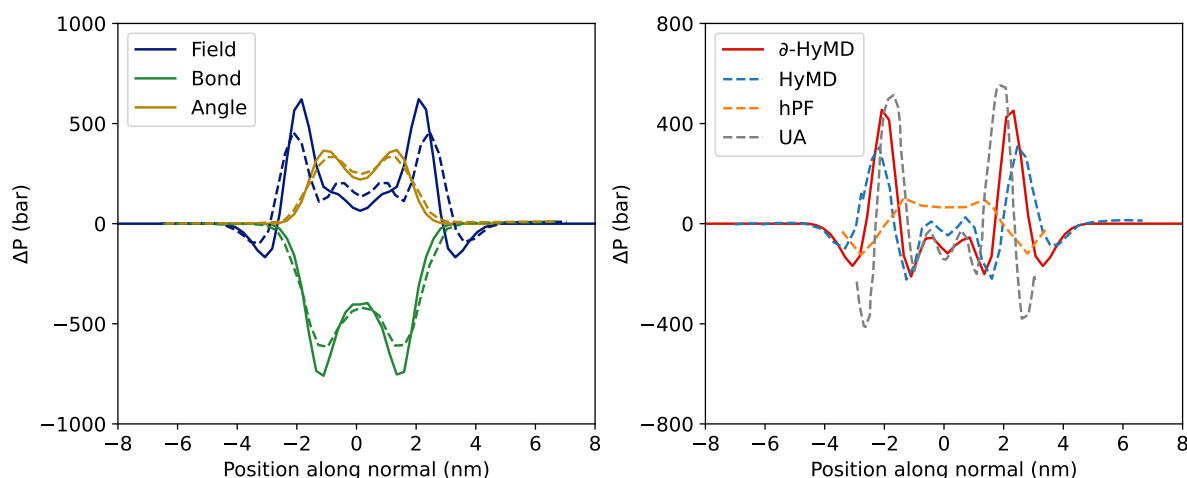


Figure 5: On the left, contributions to the pressure difference in the normal and lateral directions to the membrane ( $\Delta P$ ), across a DPPC bilayer simulated with the parameters optimized in this paper (solid line) and with previous BO parameters from Sen et al.<sup>23</sup> (dashed line) due to field, bond and angle terms. On the right, total pressure differences calculated in this work (solid line), and HhPF, hPF, and united atom (UA) curves are taken from ref<sup>23</sup>.

negative fluctuations in the outer part of the bilayer, similar to the BO model. As already discussed there, this is attributed to the poor representation of the solvation structure, which is an intrinsic weakness of the CG mapping itself. However, compared to the BO model, in this case, we managed to almost perfectly match the inner all-atom peaks, both in terms of intensity and position along the normal to the membrane. The qualitatively correct behavior of the lateral pressure profile across the whole membrane validates the soundness of the physics represented by the models, and thus their reliability for future use in other application studies. Fig. S2 in the Supporting Information reports the complete table of lateral pressure profiles for all the other systems.

Another validation we carry out is the ability of lipids to self-assemble. This was already investigated in a previous paper,<sup>9</sup> but it is an important test to replicate here, since the optimized parameters are strikingly different compared to the ones used previously. In particular, the tail bead-water interaction, which is one of the main promoters of lipid aggregation, dropped from  $42.24 \text{ kJ mol}^{-1}$  to  $23.94 \text{ kJ mol}^{-1}$ . In Fig. 6, we report simulation

snapshots that show how the new model still manages to reproduce extremely fast self-assembly of a DPPC bilayer, as characteristic of the HhPF model.

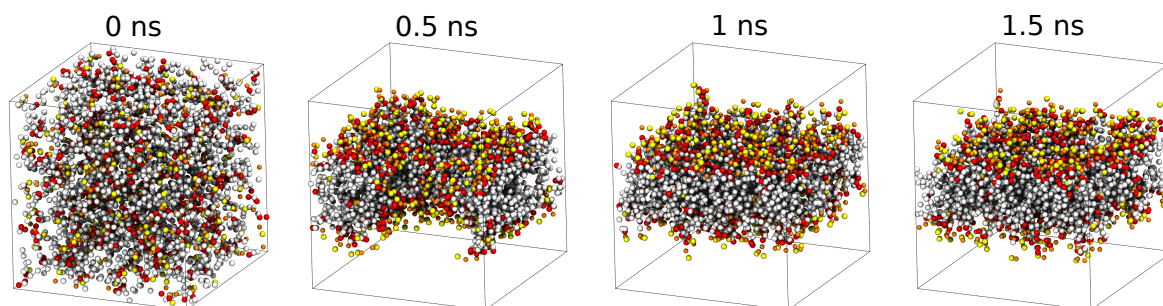


Figure 6: DPPC bilayer self-assembly with the optimized parameters in a  $10^3 \text{ nm}^3$  box. We use the same color coding shown in the mapping. Water beads are omitted for clarity.

Finally, we test the phase separation of TG molecules in a DOPC bilayer, that was previously investigated by molecular dynamics simulations and experiments.<sup>56</sup> We use TG here because the mapping does not introduce any new bead type, so we can also verify the transferability of the optimization on molecules that were not included in the training dataset. In our simulation, we use a concentration of TG that is above the reported critical aggregation concentration of 4%<sup>67</sup>, therefore we expect to observe phase separation from DOPC. In Fig.7 we report snapshots from the initial configuration and after 30 ns. We are indeed able to observe the formation of a TG blister, with a diameter of  $\sim 18 \text{ nm}$  and a height of  $\sim 5.7 \text{ nm}$ , inside the DOPC membrane. In reality, in our simulation the process of aggregation of the TG molecules starts much earlier, around the 10 ns mark, which is anyway consistent with the previously reported blister formation time of  $\sim 25 \text{ ns}$ .<sup>56</sup>

Overall, the parameters optimized for the whole set of lipids (Table S1 in the Supporting Information) perform just slightly worse than the parameters optimized for each lipid individually (Tables S3 to S14 in the Supporting Information). The data for the losses, density profiles, area per lipid, and lateral pressure profiles for the individual lipids optimization are also shown in Fig. S3 to S8 in the Supporting Information. Even though the APL and density profiles for the individually optimized lipids are closer to their experimental references, the parameters were also more prone to overfitting, leading to

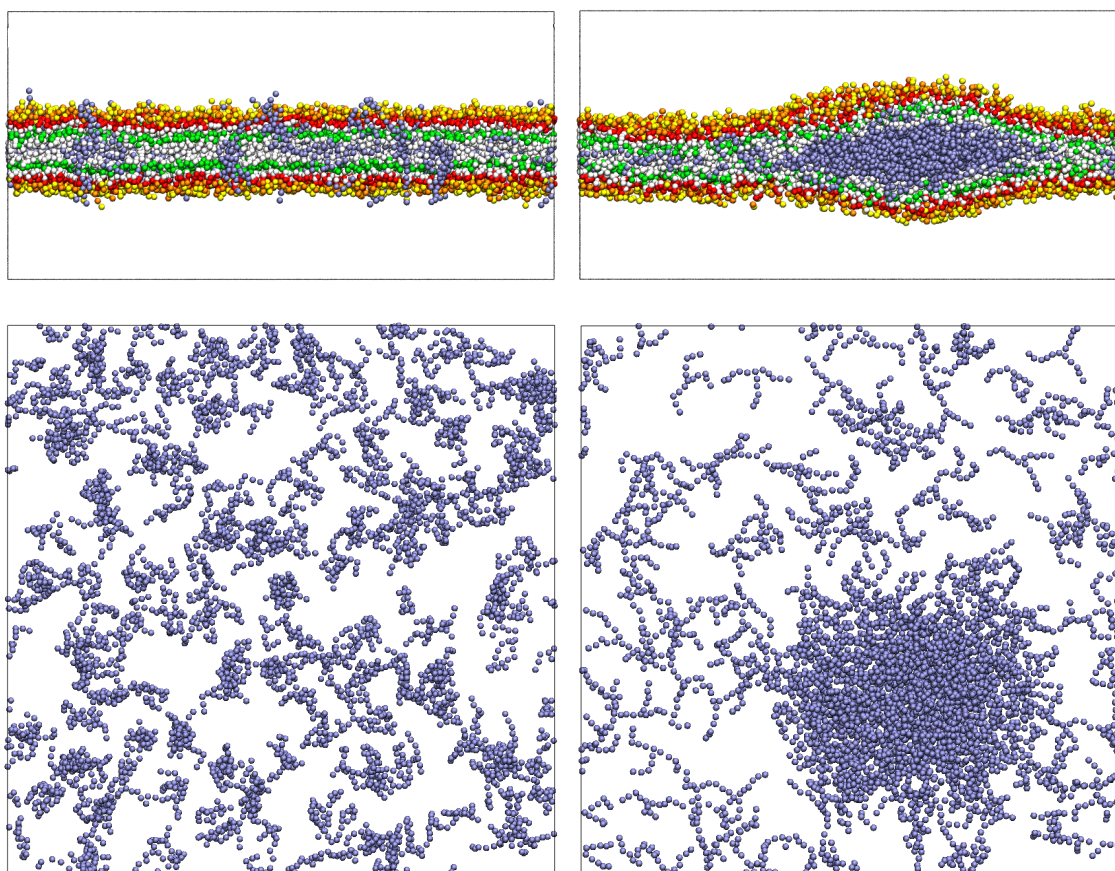


Figure 7: Phase separation of triglyceride (TG) molecules, in blue, inside a DOPC bilayer. On the left, side and top view of the starting system with TGs randomly dispersed in the membrane; on the right, snapshots taken after 30 ns that show the formation of a TG blister. DOPC molecules are not rendered in the top view, and water beads are omitted for clarity.

unstable simulations in some cases. In many cases, the parameters were also very different, so obtaining a single set of transferable parameters from this data is not trivial. Training a common set of parameters for all the lipids with the protocol presented in this work, provided transferable parameters which still lead to properties in very good agreement with the target values.



## Conclusion and Outlook

In this work, we introduced the use of differentiable molecular dynamics as a mean to automatize and standardize the determination of force field parameters for HhPF modeling. Our procedure requires the definition of a loss function, targeting specific molecular properties of interest, which may be based on any microscopic or macroscopic observable, and may target bottom-up data, i.e., from all-atom models, or top-down quantities, i.e., from experimental measurements.

Compared to similar approaches already proposed in the literature, here we overcome the general issues of excessive memory requirements, and numerical instability upon progressive gradient accumulation, by leveraging an implementation procedure, where a set of relatively short, independent parallel runs, collectively contribute to the determination of the target properties that enter into the loss function. This framework is implemented in  $\partial$ -HyMD, an open-source software that reimplements HyMD by employing the autodiff capabilities of JAX.

We tested our implementation by optimizing HhPF models for a library of phospholipids, comprising different polar heads (phosphocholine, phosphoethylamine, phosphoglycerol), as well as different combinations of saturated and unsaturated fatty tails. The results show that, in general, our differentiable MD protocol can systematically optimize the parameter sets, producing models that are in excellent agreement with the reference. In particular, the model produced by differentiable MD for DPPC surpasses the one previously obtained by BO, showing better-peaked distributions for the lipid heads, and improved water penetration at the water/lipid interface, even though the parameters are optimized to be transferable to other lipids in the  $\partial$ -HyMD case and were tailored to just DPPC in the BO case. These considerations indicate differentiable MD as a more solid route for the systematic optimization of libraries of compounds.

The parameters obtained by  $\partial$ -HyMD reproduce with good accuracy the fluctuations in the difference between the normal and lateral components of the pressure, as compared to

the all-atom reference, and previous gradient-free parameterization. In particular, they yield the tensionless condition for the bilayer as a balance between the laterally compacting hydrophobic tails, and the laterally expanding hydrophilic heads. The good quality of the pressure profiles confirms that the parameters obtained by this optimization procedure describe well the physics of the system, and can be used to predict properties not explicitly introduced into the learning pool. We also showed that the fast self-assembly of lipid membranes is retained by these parameters, and that the extension of their use for molecules outside the training pool, such as triglycerides, is possible.

In this work, we also introduced the first library of standard charged and zwitterionic phospholipids to be used with the HhPF method. In fact, our  $\partial$ -HyMD is, by design, a flexible tool that can be easily applied to diverse systems. Future work will be focused on the enrichment of this core library with other biologically relevant lipids, including sphingolipids, sterols, and glycosylated systems (phosphorylated inositol phospholipids, lipopolysaccharides). A further step will be in the use of this same approach for the calibration of sequence-specific peptide models,<sup>68</sup> and of nucleic acids, with the final objective of producing a consistent force-field for simulations of complex multiphase biological systems. Changing the mapping and including the optimization of the bonded parameters may also lead to improvements in the description of membranes.<sup>61</sup>

## Acknowledgments

This work was supported by the Research Council of Norway through the Centre of Excellence *Hylleraas Centre for Quantum Molecular Sciences* (grant number 262695), by the Norwegian Supercomputing Program (NOTUR) (grant number NN4654K), and by the Deutsche Forschungsgemeinschaft (DFG) within the project B5 of the TRR-146 (project number 233630050).




## Data Availability


$\partial$ -HyMD is an open source software released under LGPLv3 license, freely downloadable at: <https://github.com/Cascella-Group-UiO/Diff-HyMD>. The HylleraasMD code is also released under LGPLv3 open source software license, freely downloadable at: <https://github.com/Cascella-Group-UiO/HyMD>. Input and simulations data will become freely available for download upon publication of the present manuscript, at: <https://github.com/Cascella-Group-UiO/Publications>


## Author information


### Corresponding authors


**Michele Cascella**  – Hylleraas Centre for Quantum Molecular Sciences and Department of Chemistry, University of Oslo, PO Box 1033 Blindern, 0315 Oslo, Norway; Email: [michele.cascella@kjemi.uio.no](mailto:michele.cascella@kjemi.uio.no)

### Authors

**Manuel Carrer**  – Hylleraas Centre for Quantum Molecular Sciences and Department of Chemistry, University of Oslo, PO Box 1033 Blindern, 0315 Oslo, Norway

**Henrique Musseli Cezar**  – Hylleraas Centre for Quantum Molecular Sciences and Department of Chemistry, University of Oslo, PO Box 1033 Blindern, 0315 Oslo, Norway

**Morten Ledum**  – Hylleraas Centre for Quantum Molecular Sciences and Department of Chemistry, University of Oslo, PO Box 1033 Blindern, 0315 Oslo, Norway

**Sigbjørn Løland Bore**  – Hylleraas Centre for Quantum Molecular Sciences and Department of Chemistry, University of Oslo, PO Box 1033 Blindern, 0315 Oslo, Norway

### Notes

The authors declare no competing financial interest.

## References

- (1) Voth, G. A. Coarse-Graining of Condensed Phase and Biomolecular Systems; CRC press, 2008.
- (2) Kmiecik, S.; Gront, D.; Kolinski, M.; Wieteska, L.; Dawid, A. E.; Kolinski, A. Coarse-Grained Protein Models and Their Applications. Chem. Rev. **2016**, 116, 7898–7936.
- (3) Klein, M. L.; Shinoda, W. Large-scale molecular dynamics simulations of self-assembling systems. Science **2008**, 321, 798–800.
- (4) Saunders, M. G.; Voth, G. A. Coarse-Graining Methods for Computational Biology. Annu. Rev. Biophys. **2013**, 42, 73–93.
- (5) Shelley, J. C.; Shelley, M. Y.; Reeder, R. C.; Bandyopadhyay, S.; Klein, M. L. A coarse grain model for phospholipid simulations. J. Phys. Chem. B **2001**, 105, 4464–4470.
- (6) Shelley, J. C.; Shelley, M. Y.; Reeder, R. C.; Bandyopadhyay, S.; Moore, P. B.; Klein, M. L. Simulations of phospholipids using a coarse grain model. J. Phys. Chem. B **2001**, 105, 9785–9792.
- (7) Molinero, V.; Moore, E. B. Water Modeled as an Intermediate Element between Carbon and Silicon. J. Phys. Chem. B **2009**, 113, 4008–4016.
- (8) Daoulas, K. C.; Müller, M. Single Chain in Mean Field Simulations: Quasi-instantaneous Field Approximation and Quantitative Comparison with Monte Carlo Simulations. J. Chem. Phys. **2006**, 125, 184904.
- (9) Ledum, M.; Sen, S.; Li, X.; Carrer, M.; Feng, Y.; Cascella, M.; Bore, S. L. HylleraasMD: A Domain Decomposition-Based Hybrid Particle-Field Software for Multiscale Simulations of Soft Matter. J. Chem. Theory Comput. **2023**, 19, 2939–2952.

- (10) Bjørnestad, V. A.; Li, X.; Tribet, C.; Lund, R.; Cascella, M. Micelle Kinetics of Photoswitchable Surfactants: Self-assembly Pathways and Relaxation Mechanisms. J. Colloid Interface Sci. **2023**, 646, 883–899.
- (11) Marrink, S. J.; de Vries, A. H.; Mark, A. E. Coarse Grained Model for Semiquantitative Lipid Simulations. J. Phys. Chem. B **2004**, 108, 750–760.
- (12) Marrink, S. J.; Risselada, H. J.; Yefimov, S.; Tieleman, D. P.; de Vries, A. H. The MARTINI Force Field: Coarse Grained Model for Biomolecular Simulations. J. Phys. Chem. B **2007**, 111, 7812–7824.
- (13) Souza, P. C.; Alessandri, R.; Barnoud, J.; Thallmair, S.; Faustino, I.; Grünewald, F.; Patmanidis, I.; Abdizadeh, H.; Bruininks, B. M.; Wassenaar, T. A., et al. Martini 3: a general purpose force field for coarse-grained molecular dynamics. Nat. Methods **2021**, 18, 382–388.
- (14) Bore, S. L.; Cascella, M. Hamiltonian and Alias-Free Hybrid Particle–Field Molecular Dynamics. J. Chem. Phys. **2020**, 153, 094106.
- (15) Ledum, M.; Sen, S.; Bore, S. L.; Cascella, M. On the Equivalence of the Hybrid Particle–Field and Gaussian Core Models. J. Chem. Phys. **2023**, 158, 194902.
- (16) Schneider, L.; Müller, M. Multi-Architecture Monte-Carlo (MC) Simulation of Soft Coarse-Grained Polymeric Materials: SOft Coarse Grained Monte-Carlo Acceleration (SOMA). Comput. Phys. Commun. **2019**, 235, 463–476.
- (17) Ledum, M.; Carrer, M.; Sen, S.; Li, X.; Cascella, M.; Bore, S. L. HylleraasMD: Massively Parallel Hybrid Particle-Fieldmolecular Dynamics in Python. J. Open Source Softw. **2023**, 8, 4149.
- (18) Reith, D.; Pütz, M.; Müller-Plathe, F. Deriving Effective Mesoscale Potentials from Atomistic Simulations. J. Comput. Chem. **2003**, 24, 1624–1636.

- (19) Noid, W. G.; Chu, J.-W.; Ayton, G. S.; Krishna, V.; Izvekov, S.; Voth, G. A.; Das, A.; Andersen, H. C. The Multiscale Coarse-Graining Method. I. A Rigorous Bridge between Atomistic and Coarse-Grained Models. J. Chem. Phys. **2008**, 128, 244114.
- (20) De Nicola, A.; Zhao, Y.; Kawakatsu, T.; Roccatano, D.; Milano, G. Hybrid Particle-Field Coarse-Grained Models for Biological Phospholipids. J. Chem. Theory Comput. **2011**, 7, 2947–2962.
- (21) Ledum, M.; Bore, S. L.; Cascella, M. Automated Determination of Hybrid Particle-Field Parameters by Machine Learning. Mol. Phys. **2020**, 118, e1785571.
- (22) Seifrid, M.; Pollice, R.; Aguilar-Granda, A.; Morgan Chan, Z.; Hotta, K.; Ser, C. T.; Vestfrid, J.; Wu, T. C.; Aspuru-Guzik, A. Autonomous Chemical Experiments: Challenges and Perspectives on Establishing a Self-Driving Lab. Acc. Chem. Res. **2022**, 55, 2454–2466.
- (23) Sen, S.; Ledum, M.; Bore, S. L.; Cascella, M. Soft Matter under Pressure: Pushing Particle-Field Molecular Dynamics to the Isobaric Ensemble. J. Chem. Inf. Model. **2023**, 63, 2207–2217.
- (24) Frazier, P. I. A Tutorial on Bayesian Optimization. arXiv **2018**, arXiv:1807.02811.
- (25) Baydin, A. G.; Pearlmutter, B. A.; Radul, A. A.; Siskind, J. M. Automatic Differentiation in Machine Learning: A Survey. arXiv **2015**, arXiv:1502.05767.
- (26) Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. Learning Representations by Back-Propagating Errors. Nature **1986**, 323, 533–536.
- (27) Schoenholz, S.; Cubuk, E. D. JAX MD: A Framework for Differentiable Physics. Advances in Neural Information Processing Systems. 2020; pp 11428–11441.
- (28) Wang, Y.; Fass, J.; Kaminow, B.; Herr, J. E.; Rufa, D.; Zhang, I.; Pulido, I.; Henry, M.;

- Bruce Macdonald, H. E.; Takaba, K.; Chodera, J. D. End-to-end differentiable construction of molecular mechanics force fields. Chem. Sci. **2022**, 13, 12016–12033.
- (29) Wang, W.; Wu, Z.; Dietschreit, J. C. B.; Gómez-Bombarelli, R. Learning Pair Potentials Using Differentiable Simulations. J. Chem. Phys. **2023**, 158, 044113.
- (30) Wang, W.; Axelrod, S.; Gómez-Bombarelli, R. Differentiable Molecular Simulations for Control and Learning. arXiv **2020**, arXiv:2003.00868.
- (31) Thaler, S.; Zavadlav, J. Learning Neural Network Potentials from Experimental Data via Differentiable Trajectory Reweighting. Nat. Commun. **2021**, 12, 6884.
- (32) Doerr, S.; Majewski, M.; Pérez, A.; Krämer, A.; Clementi, C.; Noe, F.; Giorgino, T.; De Fabritiis, G. TorchMD: A Deep Learning Framework for Molecular Simulations. J. Chem. Theory Comput. **2021**, 17, 2355–2363.
- (33) Greener, J. G.; Jones, D. T. Differentiable Molecular Simulation Can Learn All the Parameters in a Coarse-Grained Force Field for Proteins. PLoS ONE **2021**, 16, e0256990.
- (34) Bradbury, J.; Frostig, R.; Hawkins, P.; Johnson, M. J.; Leary, C.; Maclaurin, D.; Necula, G.; Paszke, A.; VanderPlas, J.; Wanderman-Milne, S.; Zhang, Q. JAX: Autograd and XLA. Astrophysics Source Code Library, record ascl:2111.002, 2021.
- (35) Häfner, D.; Vicentini, F. mpi4jax: Zero-copy MPI communication of JAX arrays. Journal of Open Source Software **2021**, 6.
- (36) Milano, G.; Kawakatsu, T. Hybrid Particle-Field Molecular Dynamics Simulations for Dense Polymer Systems. J. Chem. Phys. **2009**, 130, 214106.
- (37) Vogiatzis, G. G.; Megariotis, G.; Theodorou, D. N. Equation of State Based Slip Spring Model for Entangled Polymer Dynamics. Macromolecules **2017**, 50, 3004–3029.

- (38) Hockney, R. W.; Eastwood, J. W. Computer Simulation Using Particles; crc Press, 2021.
- (39) Tuckerman, M.; Berne, B. J.; Martyna, G. J. Reversible Multiple Time Scale Molecular Dynamics. J. Chem. Phys. **1992**, 97, 1990–2001.
- (40) Schoenholz, S. S.; Cubuk, E. D. JAX, M.D. A framework for differentiable physics\*. J. Stat. Mech. Theory Exp. **2021**, 2021, 124016.
- (41) Molly. **2023**, JuliaMolSim, <https://github.com/JuliaMolSim/Molly.jl>.
- (42) Wu, E. L.; Cheng, X.; Jo, S.; Rui, H.; Song, K. C.; Dávila-Contreras, E. M.; Qi, Y.; Lee, J.; Monje-Galvan, V.; Venable, R. M.; Klauda, J. B.; Im, W. CHARMM-GUI Membrane Builder toward Realistic Biological Membrane Simulations. J. Comput. Chem. **2014**, 35, 1997–2004.
- (43) Huang, J.; Rauscher, S.; Nawrocki, G.; Ran, T.; Feig, M.; de Groot, B. L.; Grubmüller, H.; MacKerell, A. D. CHARMM36m: An Improved Force Field for Folded and Intrinsically Disordered Proteins. Nat. Methods **2017**, 14, 71–73.
- (44) Berendsen, H. J. C.; van der Spoel, D.; van Drunen, R. GROMACS: A message-passing parallel molecular dynamics implementation. Comput. Phys. Commun. **1995**, 91, 43–56.
- (45) Abraham, M. J.; Murtola, T.; Schulz, R.; Páll, S.; Smith, J. C.; Hess, B.; Lindahl, E. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. SoftwareX **2015**, 1-2, 19–25.
- (46) Bussi, G.; Donadio, D.; Parrinello, M. Canonical Sampling through Velocity Rescaling. J. Chem. Phys. **2007**, 126, 014101.
- (47) Bernetti, M.; Bussi, G. Pressure Control Using Stochastic Cell Rescaling. J. Chem. Phys. **2020**, 153, 114107.

- (48) Hess, B.; Bekker, H.; Berendsen, H. J. C.; Fraaije, J. G. E. M. LINCS: A Linear Constraint Solver for Molecular Simulations. J. Comput. Chem. **1997**, 18, 1463–1472.
- (49) Graham, J. A.; Essex, J. W.; Khalid, S. PyCGTOOL: Automated Generation of Coarse-Grained Molecular Dynamics Models from Atomistic Trajectories. J. Chem. Inf. Model. **2017**, 57, 650–656.
- (50) Marrink, S. J.; de Vries, A. H.; Mark, A. E. Coarse Grained Model for Semiquantitative Lipid Simulations. J. Phys. Chem. B **2004**, 108, 750–760.
- (51) Michaud-Agrawal, N.; Denning, E. J.; Woolf, T. B.; Beckstein, O. MDAAnalysis: A Toolkit for the Analysis of Molecular Dynamics Simulations. J. Comput. Chem. **2011**, 32, 2319–2327.
- (52) Gowers, R.; Linke, M.; Barnoud, J.; Reddy, T.; Melo, M.; Seyler, S.; Domański, J.; Dotson, D.; Buchoux, S.; Kenney, I.; Beckstein, O. MDAAnalysis: A Python Package for the Rapid Analysis of Molecular Dynamics Simulations. Python in Science Conference. Austin, Texas, 2016; pp 98–105.
- (53) Berendsen, H. J. C.; Postma, J. P. M.; Van Gunsteren, W. F.; DiNola, A.; Haak, J. R. Molecular Dynamics with Coupling to an External Bath. J. Chem. Phys. **1984**, 81, 3684–3690.
- (54) Zhuang, J.; Tang, T.; Ding, Y.; Tatikonda, S.; Dvornek, N.; Papademetris, X.; Duncan, J. S. AdaBelief Optimizer: Adapting Stepsizes by the Belief in Observed Gradients. arXiv **2020**, arXiv:2010.07468.
- (55) Vuorela, T.; Catte, A.; Niemelä, P. S.; Hall, A.; Hyvönen, M. T.; Marrink, S.-J.; Karttunen, M.; Vattulainen, I. Role of Lipids in Spheroidal High Density Lipoproteins. 6, e1000964.

- (56) Zoni, V.; Khaddaj, R.; Campomanes, P.; Thiam, A. R.; Schneider, R.; Vanni, S. Pre-Existing Bilayer Stresses Modulate Triglyceride Accumulation in the ER versus Lipid Droplets. 10, e62886.
- (57) Jaschonek, S.; Cascella, M.; Gauss, J.; Diezemann, G.; Milano, G. Intramolecular structural parameters are key modulators of the gel-liquid transition in coarse grained simulations of DPPC and DOPC lipid bilayers. Biochem. Biophys. Res. Commun. **2018**, 498, 327–333.
- (58) Nagle, J. F.; Tristram-Nagle, S. Structure of Lipid Bilayers. 1469, 159–195.
- (59) Petrache, H. I.; Dodd, S. W.; Brown, M. F. Area per Lipid and Acyl Length Distributions in Fluid Phosphatidylcholines Determined by <sup>2</sup>H NMR Spectroscopy. Biophys. J. **2000**, 79, 3172–3192.
- (60) Kucerka, N.; Nieh, M.-P.; Katsaras, J. Fluid Phase Lipid Areas and Bilayer Thicknesses of Commonly Used Phosphatidylcholines as a Function of Temperature. Biochim. Biophys. Acta - Biomembr. **2011**, 1808, 2761–2771.
- (61) Empereur-mot, C.; Pedersen, K. B.; Capelli, R.; Crippa, M.; Caruso, C.; Perrone, M.; Souza, P. C. T.; Marrink, S. J.; Pavan, G. M. Automatic Optimization of Lipid Models in the Martini Force Field Using SwarmCG. J. Chem. Inf. Model. **2023**, 63, 3827–3838.
- (62) Kucerka, N.; Nagle, J. F.; Sachs, J. N.; Feller, S. E.; Pencer, J.; Jackson, A.; Katsaras, J. Lipid Bilayer Structure Determined by the Simultaneous Analysis of Neutron and X-Ray Scattering Data. Biophys. J. **2008**, 95, 2356–2367.
- (63) Nagle, J.; Wiener, M. Structure of Fully Hydrated Bilayer Dispersions. Biochim. Biophys. Acta - Biomembr. **1988**, 942, 1–10.
- (64) Gawrisch, K.; Parsegian, V. A.; Hajduk, D. A.; Tate, M. W.; Gruner, S. M.; Fuller, N. L.; Rand, R. P. Energetics of a Hexagonal-Lamellar-Hexagonal-Phase



- Transition Sequence in Dioleoylphosphatidylethanolamine Membranes. Biochemistry **1992**, 31, 2856–2864.
- (65) Rand, R.; Parsegian, V. Hydration Forces between Phospholipid Bilayers. Biochim. Biophys. Acta - Rev. Membr. **1989**, 988, 351–376.
- (66) Pan, J.; Heberle, F. A.; Tristram-Nagle, S.; Szymanski, M.; Koepfinger, M.; Katsaras, J.; Kucerka, N. Molecular Structures of Fluid Phase Phosphatidylglycerol Bilayers as Determined by Small Angle Neutron and X-ray Scattering. Biochim. Biophys. Acta - Biomembr. **2012**, 1818, 2135–2148.
- (67) Khandelia, H.; Duelund, L.; Pakkanen, K. I.; Ipsen, J. H. Triglyceride Blisters in Lipid Bilayers: Implications for Lipid Droplet Biogenesis and the Mobile Lipid Signal in Cancer Cell Membranes. 5, e12811.
- (68) Bore, S. L.; Milano, G.; Cascella, M. Hybrid particle-field model for conformational dynamics of peptide chains. J. Chem. Theory Comput **2018**, 14, 1120–1130.

# TOC Graphic

